THE GENERATION OF FRENCH FROM A SEMANTIC REPRESENTATION

Annette Herskovits

Stanford University

Prepared for:

Advanced Research Projects Agency

August 1973

DISCLAIMER NOTICE

THIS DOCUMENT IS THE BEST QUALITY AVAILABLE.

COPY FURNISHED CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Unclassified
Security Classification

AD-769379

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)
Stanford University
Computer Science Department
Stanford, California, 94305

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP
Blank

3. REPORT TITLE
The generation of French from a semantic representation

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Technical report, August 1973

5. AUTHOR(S) (First name, middle initial, last name)
Annette Herskovits

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
August 1973	~~28~~ 23	3

8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)
SD 183	STAN-CS 73-384
b. PROJECT NO.	
c. 457	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	AIM - 212

10. DISTRIBUTION STATEMENT
Releasable without limitations on dissemination

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
Blank	Blank

13. ABSTRACT

The report contains first a brief description of Preference Semantics, a system of representation and analysis of the meaning structure of natural language. The analysis algorithm which transforms phrases into semantic items called templates has been considered in detail elsewhere, so this report concentrates on the second phase of analysis, which binds templates together into a higher level semantic block corresponding to an English paragraph, and which, in operation, interlocks with the French generation procedure. During this phase, the semantic relations between templates are extracted, pronouns are referred and those word disambiguations are done that require the context of a whole paragraph. These tasks require items called PARAPLATES which are attached to keywords such as prepositions, subjunctions and relative pronouns. The system chooses the representation which maximises a carefully defined "semantic density."

A system for the generation of French sentences is then described, based on the recursive evaluation of procedural generation patterns called STEREOTYPES. The stereotypes are semantically context sensitive, are attached to each sense of English words and keywords and are carried into the representation by the analysis procedure. The representation of the meaning of words, and the versatility of the stereotype format, allow for fine meaning distinctions to appear in the French, and for the construction of French differing radically from the English original.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6801

Security Classification

STANFORD ARTIFICIAL INTELLIGENCE LABORATORY AUGUST 1973
MEMO NO. AIM-212

COMPUTER SCIENCE DEPARTMENT
REPORT NO. CS-384

THE GENERATION OF FRENCH FROM A SEMANTIC REPRESENTATION

by

ANNETTE HERSKOVITS

ABSTRACT: The report contains first a brief description of Preference Semantics, a system of representation and analysis of the meaning structure of natural language. The analysis algorithm which transforms phrases into semantic items called templates has been considered in detail elsewhere, so this report concentrates on the second phase of analysis, which binds templates together into a higher level semantic block corresponding to an English paragraph, and which, in operation, interlocks with the French generation procedure. During this phase, the semantic relations between templates are extracted, pronouns are referred and those word disambiguations are done that require the context of a whole paragraph. These tasks require items called PARAPLATES which are attached to keywords such as prepositions, subjunctions and relative pronouns. The system chooses the representation which maximises a carefully defined "semantic density".

A system for the generation of French sentences is then described, based on the recursive evaluation of procedural generation patterns called STEREOTYPES. The stereotypes are semantically context sensitive, are attached to each sense of English words and keywords and are carried into the representation by the analysis procedure. The representation of the meaning of words, and the versatility of the stereotype format, allow for fine meaning distinctions to appear in the French, and for the construction of French differing radically from the English original.

This research was supported by the Advanced Research Projects Agency, Department of Defense(SD 183), USA.

CHAPTER I

INTRODUCTION

This paper describes the generation of French sentences from a semantic representation of natural language conceived by Yorick Wilks [1]. The generation procedure is part of a system which takes as input English paragraphs, transforms them into an Interlingual representation (IR) and outputs a French translation. The system, called Preference Semantics, differs from former earlier attempts to do machine translation (MT), in that it involves no explicit syntactical analysis, but uses instead semantic means at every level of analysis and generation. In fact, the system can be said to "understand" the text translated.

Preference Semantics is characterized by:

1) lexical decomposition. Each sense of a word of the source language is coded by a tree of semantic markers or elements from a finite set of fundamental concepts. This structure is called a "semantic formula".

2) it involves a catalogue of case relationships , such as: actor ↔ action, event ↔ location. Their occurence in a text is made explicit; thus, an English sentence is transformed into a network of lexical decomposition tree, where the arcs represent case relationships .

3) the network is organised on two levels: at the lower level are templates corresponding to fragments of English (what constitutes a fragment will be made precise later but correponds to the concept of a phrase). The templates in turn are organised into a higher level network. The analysis routines proceed in two stages corresponding to these two levels of organisation. First the text is fragmented and the semantic analysis carried out within the context of a fragment. Then, a second stage deals with semantic relations between fragments, including the referral of pronouns.

4) At each stage, the system directs itself toward the correct repesentation by preferring the most "semantically dense" one: that is, as a somewhat crude approximation, the one such that the redundancy among the lexical decomposition trees is largest.

We feel that lexical decomposition together with this method of selection of the right meaning for a sentence constitute a reasonable formalization of the representation humans maintain in their memory and of the process they carry out when they understand language. Introspective observation brings intuitive support to the fact that, whatever complex mental object is associated with a given word sense,

understanding a sentence involves "intersecting" those representations. Thus if we say "I hear a bark", the right interpretation arises because the mental objects associated respectively with "hear" and with "bark" as an animal cry, intersect extensively, whereas tree coverings and sounds cannot be connected in an immediate way. We are convinced that such semantic connections are used to establish the meaning of an utterance prior to any grammatical analysis.

Clearly the mental image associated with a word is a very complex memory item involving sensory as well as symbolic elements. But a network of fundamental concepts seems a reasonably good map for it, in terms of the "understanding" performance which an algorithm working on a "maximum intersection" principle can achieve with it, as we will see.

Lexical decomposition is one form of a data base of knowledge about the world and some general inference making mechanism could plausibly do the work of the Preference Semantics method of meaning selection. However, the major part of understanding relies on intelligent use of semantic information which can be made available in adequate lexical decomposition. This recommends that this information be coded in the most economical way, that it be readily accessible without time consuming search. Preference Semantics seems a most natural and effective way of meeting these requirements.

However, there are some cases when a correct English-French translation requires the knowlege of facts not naturally expressed in lexical decomposition, and a way of inferring from the text and from this store of knowledge. Here is an example:

The soldiers fired at the women; I saw them stagger and fall.

Referring the pronoun "them" in the second sentence would require some equivalent of the following "reasoning": firing at someone usually wounds him; wounded people often loose balance and stagger; thus "them" refers to "the women". The first fact would logically appear in the lexical decompositon of "fire at"; i.e. the purpose of "firing at" is usually to hurt. But the rest involves knowledge that could not be reasonably coded within the semantic formulas of the words occurring in the sentence.

Thus we are in the process of adding to the system a component called Common Sense Inferences, which is conceived as a natural extension of the existing Preference Semantics system, inthat it uses the same formalism and preference principle (Wilks [2]).

Two other problems involved in correctly translating English into French require machinery of another kind.

1) Consider "I drink wine" and " I like wine". In the first case we have in French "du vin" and in the second "le vin (a finite quantity of wine versus wine as a substance).

2) "I went for a walk this morning" and "I went for a walk every morning" give respectively: "Je me suis promenee ce matin" and "Je me promenais tous les matins". The imperfect is used in French for a repetitive action and the past for a one-time action.

Although in principle, questions such as "are we concerned here with wine as a species" or "is this action habitual" could be answered by using the inference mechanism, they are too complex to be dealt with in this way in practice. Thus we will implement special semantic procedures which will use the semantic representation together with some heuristics to answer these specific questions.

Correct translation from one language into another is one test of "understanding" for a computer system. Questions about whether systems capable of carrying out an "intelligent" conversation exhibit more "understanding" are meaningless without first defining in some precise way the class of questions which they are able to answer. This being rather a difficult problem, we will simply note that with the Inference component, which is at this time already precisely defined and being programmed, nothing significant has to be added to the system to extend it into a question answering system which will answer a non trivial class of questions. It remains to define "non trivial" more precisely and to compare the performance of such a system with other question answering systems.

Wilks has described in detail the semantic representation and the first stage of the analysis (Wilks [1]); we will thus present here only a brief description of both with particular attention to aspects relevant to the generation procedure. We will then describe in detail the second stage of analysis (i.e. the interfragment analysis) and the French generation routines as they are both conceptually and programmatically intertwined.

CHAPTER II

THE INTERLINGUA AND INTERNAL ANALYSIS OF FRAGMENTS.

We will first describe the interlingual building blocks or ELEMENTS, then each significant substructure of the Interlingua together with the various procedures which constitute the intrafragment analysis. We will describe the final outlook of the IR, but will consider the interfragment analysis only in the next chapter.

ELEMENTS

They are 60 semantic primitives corresponding to fundamental concepts and relations. Here are some examples (in capital letters) followed by a discursive description:

(a) entities:
MAN (human beings), STUFF (substances), THING (physical object) etc...

(b) actions:
HAVE (possesses), FORCE (compels), CAUSE (causes to happen) etc...

(c) type indicators:
KIND (being a quality), HOW (being a type of action) etc..

(d) sorts:
WHOLE (being a totality), GOOD (being morally acceptable), THRU (being an aperture) etc...

(e) cases:
AT (location), WITH (instrument), SUBJ (agent), OBJE (patient of action), IN (containment), POSS (possessed by) etc...

FORMULAS

A semantic formula is a binary tree structure of ELEMENTS, expressing the semantic content of a concept. In our dictionary, each sense of an English word is coded with such a formula. For example:

((*ANI SUBJ)((*ANI OBJE)(((LIFE OBJE) NOTHAVE) CAUSE)))

represents the meaning of 'to kill".

At any fork of the binary tree, there is a dependency relation of the left branch upon the right branch. This dependency is interpreted differently but unambiguously, according to the left and right subtrees: for example to the left of CAUSE, we expect to find a subformula referring to what has been caused. The subformula with OBJE as a right member indicates the class of preferred objects of the action, here *ANI or class of animate beings. Similarly, (*ANI SUBJ) indicates that the subject of "to kill" is generally an animate being. Thus the whole formula says that "to kill" is " an animate being causing an animate being to lose life".

A consequence of the left to right dependency rule is that the rightmost element of a formula, the HEAD, is the primitive whose semantic scope comprehends most adequately that of the concept described by the formula. The choice of a head for a given concept is sometimes debatable.

For example, one sense of "to urge" has been coded:

((MAN SUBJ)((*ANI OBJE)(FORCE TELL)))

The head is TELL, which means "to communicate verbally"; in trying to define "to urge, " this might be the first delimitation of the meaning we would like to do, given the choice of primitives that is available to us. However we might prefer FORCE as a head, with the rightmost subformula (TELL FORCE), thinking of "to urge" as "to encourage verbally" rather than "to utter encouragements". The decision is largely dependent, as is the whole coding and even the basic dicrimination of word senses, on the task which we set ourselves with the interlingual representation. We will come back on this point later, when speaking specifically of problems of translation into French.

More details about the syntax and semantics of formulas is available in Wilks [1].

BARE TEMPLATES

A bare template is an ordered triple of elements, whose semantic interdependence is that of an agent-act-object triple. Our inventory of bare templates should contain all and only those triples which can be built as follows: by aligning the heads of the formulas of the agent, action and object of any natural language statement which does not involve nonsense or metaphors. Thus MAN_GIVE_THING is a bare template, but not MAN-BE-THING (the semantic scope of the elements GIVE and BE should be obvious). Presumably, no statement respecting the above restrictions would have for core of meaning "a man is a physical object"; but "John offered a motocycle to his son" yields the bare template MAN-GIVE-THING. The significance of bare templates lies in the way in which they function in the analysis algorithm, which we will now sketch.

FRAGMENTATION

The original English text is first fragmented: at punctuation marks; keywords such as subjunctions, prepositions, connectives and relative pronouns; before gerunds and where "that" has been omitted.

BARE TEMPLATE MATCHING

As we have seen, to each English word in our dictionary is attached one or more formulas corresponding to the various senses of the word. Working within the context of single fragment, we form all sequences of formulas which can be obtained, by picking for each word of the fragment, one of its formulas. The corresponding sequence of heads is then examined; if three heads, not necessarily consecutive but in the order of the corresponding text, make a triple which is in the bare bare template inventory, then we keep the corresponding sequence of formulas for further examination; otherwise, this "interpretation" of the fragment is eliminated. Thus bare template matching is a tool 1) for cutting down the number of interpretations of the words in the fragment, 2) for making a first grammatical analysis.

For example: "Small men sometimes father big sons" will give the two sequences of heads:

KIND MAN HOW MAN KIND MAN

and

KIND MAN HOW CAUSE KIND MAN.

(CAUSE is the head of the verbal sense of "father": "to father" is analyzed as "to cause to have life".)

The first sequence has no underlying bare template; however, in the second we find MAN-CAUSE-MAN which is a legitimate bare template. Thus we have disambiguated "father". At the same time it proposes one or several plausible agent-action-object substructures.

However, as not all fragments follow an actor-act-object pattern we have extended our inventory of bare templates as follows:

1)we use dummy elements as place-holders for missing items, OTHIS for the actor and object places, and OBE in the act place. Thus THING-OBE-OTHIS and MAN-MOVE-OTHIS are legitimate bare templates.

2) we consider that prepositions carry a verbal meaning; thus they are coded by formulas with heads POO (for "to", "into", "from" etc..) or PBE ("in", "at" ...) which occupy the center place in the relevant bare templates. This yields bare templates such as: OTHIS-PBE-POINT, OTHIS-POO-THING which would be matched respectively upon phrases like "at the crossroad" and "out of the

box" (POINT refers to point-like entities in space or time).

TEMPLATES

The process just described has selected a certain number of formula triples, which we will refer to as the templates for the fragment.

EXPANSION

The expansion algorithm 1) carries through disambiguation as far as the context of a fragment permits; 2) performs the work of a conventional grammar: namely it makes explicit linguistic dependencies such as that of agent on act, indirect object on act, qualifier on substantive, etc...

Expansion simply means taking the one or more templates selected by the preceding matching process in the context of the fragment from which they came, and looking again at the formulas left behind, those which did not get picked up by template matching, and seeing which of them, if any, can be attached to the template structure by a system of dependencies between formulas. By "dependencies", we mean relations such as agent-act, act-indirect object, qualifying adjective-substantive, etc.. between the corresponding formulas.

Our preference principle tells us to select as the correct representation for a fragment, the most expanded or densest template: the one for which the greatest number of such dependencies can be set up. This method can yield virtually all the results of a conventional grammar, while using only relations between semantic elements.

The representation derived so far is a sequence of fragments with, matched unto each, one or several expanded templates. In addition, each keyword in the dictionary is coded with a list of PARAPLATES (described in the next chapter) which have been carried along with the keyword into the still unfinished representation. This is what will be handed on as input for the second phase of analysis. We will now describe the final product of the overall analysis process, leaving aside for the time being the way in which it is derived.

THE LINKS AND FINAL FORM OF THE IR.

We are now concerned with relationships between templates, their definition and coding . To each expanded template is attached a link. A link consists of three items of information : the KEY, MARK and CASE.

The key is the keyword, if any, which triggered fragmentation; else it is NIL.

The mark is a list of one or several words outside the current fragment, each of which relates to the current fragment through the same dependency. The catalogue of dependencies considered includes linguistic relationships such as:

subject on predicate
governor on prepositional phrase
verb on object
verb of main clause on dependent clause
etc...

The case is a descriptive tag for these dependencies. The list of case names includes: AT (location in space or time), WITH (instrument), TO (direction), OUTOF (source), OBJE (object), etc...

Here is an example of an English sentence, fragmented, and with its key, mark, case and matching bare template:

fragment	key	mark	case	template
Some people believed	NIL	NIL	NIL	MAN-THINK-OTHIS
and said	and	(people)	PRED	OTHIS-TELL-OTHIS
that the student uprising could have led the country	that	(believed said)	OBJE	ACT-CAUSE-FOLK
into a revolution	into	(led)	TO	OTHIS-POO-ACT

The IR, in its final form consists of a sequence of fragments of the original text, with matched unto each:

- one, or sometimes several, links.
- the template, or triple of formulas, on which the bare template was matched.
-three "qualifier lists" which are lists of formulas containing the dependents upon the agent, act and object respectively.

ADJUSTMENT OF THE INTERLINGUA TO THE TASK OF TRANSLATION.

There is a class of discriminations of senses of a word which any understanding system must do: thus with "rank" in "a rank vegetation" and in "close the ranks". Outside those, distinctions are dictated by the task assigned to the understanding system. Thus Winograd's program, whose behavior requirement is that it understands and plans the execution of commands concerning the manipulation of

blocks, distinguishes two senses of "on top of": either "directly on the surface", or "somewhere above". There would be no point in making that distinction when translating into French, as the output ignores it. On the other hand, we will need to distinguish between "fish bones" and "mammals or birds bones" as the first is "arete" and the second "os".

In fact, an English word has as many semantic formulas attached to it as it has renderings into French according to context. There is no limit to the depth of embedding of formulas, so that very fine sense discriminations can be expressed, and the analysis algorithm embodies a powerful disambiguation mechanism whose shortcomings are not related to the fineness of discrimination. Thus we could translate "maintain" by "maintenir" in "maintain order"; by "entretenir" in "maintain relations"; and by "garder" in "maintain one's cool". The three formulas for "maintain" will contain as category of preferred object: respectively a type of arrangement (GRAIN), an activity (ACT), and an attitude (STATE).

A semantic category can perfectly well have a single member, which enables us to handle some idioms in a general way. For example, one formula for "to run" is: ((MAN SUBJ)((ACT OBJE)((SELF MOVE) CAUSE))) where the preferred object subformula is that of "errand" only: the French then wants "faire une course", and the generation patterns which we will describe below are written to produce this output.

Another example of a sense discrimination performed during analysis is "nearly". In "he nearly died", it becomes a verb in French: "Il a failli mourir". But "it is nearly morning" gives "c' est presque le matin". Thus "nearly" has two formulas: one indicates an adverb which qualifies actions, and the other an adverb qualifying time entities. The analysis will be able to attach "nearly" to the word it qualifies and generation patterns are written to handle the rephrasing.

CHAPTER III

THE TIE ROUTINES

The role of the TIE routines:

1) make explicit the links defined in the last section, namely the key-mark-case triples binding a whole template to others.

2) disambiguate content-words left unresolved after the expansion process. The first stage of analysis uses only the context of a fragment, whereas the TIE routines will consider the context of a whole sentence or more.

3) refer pronouns in simple cases. There is no easily defined border line between those examples which require the inference making component and those treated in the TIE routines. Any example requiring world knowledge that is not coded in the formulas, falls into the former category. However, the example "He drank wine out of a glass and it felt warm in his stomach" requires extended inferences to refer the "it", although it uses only information contained in the formulas. For more details see Wilks [2].

4) attach a generation pattern at certain points in the template sequence.

To carry out these tasks, we need a process analogous to bare template matching and to the assessment and counting of dependencies in the first phase of analysis: but for keys and their context instead of content words. However, we have adopted a different organisation; the reason is that the tasks involved require complex and varied semantic tests to be made on the context of a key. For example, discriminating between the senses of a key, not only according to case but also according to French output forms, necessitates fine and variegated semantic tests. A key has thus been coded with an ordered list of items called PARAPLATES, whose format is versatile and can include any desired semantic predicate.

PARAPLATES

A paraplate is:

<list of predicates> <case> <stereotype>

The third item is a generation form used by the generation routines and described in detail in the next section. The predicates here assume the form of a LISP function call and refer to LISP procedures. These procedures may embody any kind of test on the interlingual context of the key.

Before describing how the paraplates are used at a procedural level, let us consider, as an example, three consecutive paraplates out of the list of paraplates for the preposition "in", and the class of contexts of "in" on which each one will match:

```
1) (((OBJECT_H THING)(OBJECT_X CONT)(MARK_H MOVE (MOVE CAUSE))(MATCH1
WITH GOAL))
        TO
        ((PREOB DANS))

2) (((OBJECT_H THING)(MARK_H MOVE (MOVE CAUSE)))
        TO
        ((&PREOB DANS)))

3) (((MATCH2_HEAD) (MARK_H *DO))
        LOCA
        ((PREOB DANS)))
```

The first paraplate will match the sentence: "I put the key / in the lock".

The predicates MARK-H and OBJECT-H check upon the formulas of the mark and object of the preposition. In the first paraplate, they will be true iff the object of the preposition is a THING and if the mark is a movement verb (formula with head MOVE or rightmost subformula (MOVE CAUSE)). The predicate OBJECT_H is true iff the object of the preposition contains the element CONT, i.e. is a container.

Let us assume that, in our dictionary we have two senses of "lock", one for lock as a fastener, the other for the lock in a canal. Both locks are things satisfying ((OBJECT_H THING)) and containers satisfying ((OBJECT_H CONT)). Thus the first two predicates do not allow us to discriminate between these two senses. For this, we need MATCH1.

The predicate MATCH1 considers the object ("key") of the mark and the object of the preposition ("lock") and is true if their formulas contain an identical subformula with a rightmost element WITH or GOAL. This turns out to be the case if the formulas for "key" and "lock" are those corresponding to the senses appropriate to the sentence: these formulas express the fact that both corresponding objects serve the same purpose (GOAL), namely "to forbid the use of an opening" (or (((THRU PART)OBJE)NOTUSE))CAUSE) as it appears in the formula).

The predicate MARK_H tests the semantic formulas of prospective marks, and is used to select "put" here as the mark, as "put" has been coded with a rightmost subformula (MOVE CAUSE). Simultaneously, the directive case TO and the generation form ((&PREOB DANS)), ("dans la serrure"), are selected.

Note that the second paraplate will fit the sentence too. However, the order of paraplates and the TIE routine's operation, are such that, if a paraplate higher in the list fits, it has priority over the ones below. For this to be effective in the selection among interpretations, it is necessary that the order of paraplates reflects a degree of specificity of the class of contexts the paraplate fits. Thus a paraplate higher in the list prescribes the context more tightly than one below, unless they are mutually exclusive. This is equivalent to saying that more "dependencies" are ascertained by a higher paraplate, so that it is naturally preferred.

Consider now the sentence: "He put the number / in the table". There, only the third paraplate will fit, simultaneously selecting the numerical sort of table and not the flat wooden one. The predicate MATCH2_HEAD considers the heads of the formulas for "number" and "table" and is true if they are the same, which is true only for the correct sense of "table" (both heads being SIGN).

Finally, the sentence "I put the book / in the table" will fit both paraplate 2 and 3, giving the same sense of table in both cases, that of a flat surfaced object, but paraplate 2 will be preferred.

In addition to disambiguating, a fitting paraplate will yield a case, a mark and an adequate generation pattern.

PROGRAM OPERATION

Let us first assume that no ambiguity has been left over from the intrafragment analysis process, so that to each fragment is attached one expanded template and one only.

The core of the TIE routines consist of a set of rules written in BNF form representing the sequences of keys and template types corresponding to normal English sentences, assuming only one expanded template per fragment. There are 6 types of templates corresponding to the permitted combinations of dummy elements in the template; the class of templates with one dummy element in the subject position is subdivided into prepositional- and verbal-action templates. When those rules are used to "parse" the semantic representation, the relations between fragments appear, making it possible to assign mark and case, provided that the semantic information held in the key paraplates is simultaneously taken into account. This is done by "executing" the paraplates of the key in the course of the "parsing", when there are any.

When this operation is completed, a density coefficient is computed. This coefficient accounts for dependencies between templates such as agent-act, antecedent-relative clause, etc... : for prepositional phrases, the higher in the list the selected paraplate, the greater is the density increase. This density is used in disambiguating content-words as follows: formulas for the ambiguous words are

entered in turn in the interlingual fragment; each time, the above "parsing" is attempted. The set of formulas yielding the densest "parsing" gets selected, together with its links and stereotypes.

REFERRING PRONOUNS

Two processes are used to refer pronouns: one uses only the context of the fragment containing the pronoun to choose among possible referents, the other uses the context of a whole sentence or more.

The first procedure works as follows: the program collects syntactically plausible referents and makes a first selection using the following observation: substantives depending upon the same action through various case relationships either cannot refer to the same object, and this is a semantic impossibility, (thus the direction of an action (movement) cannot be its subject) or else a reflexive pronoun is used ("He has dedicated the book to himself").

The set of referent candidates is then ordered according to a priority based on syntactical observations such as : the function of a pronoun in its context is often the same as that of its referent in its own context. Thus in "John offered a present to Peter because he liked him", "he" is actually refers to "John" and not to "Peter". Finally the formulas of the candidates are substituted in turn for the pronoun inside the template and for each the density of dependencies is computed as during the expansion process. The formula giving the highest density or, if there are several of those, the one among them with highest priority is selected.

The second process is similar to the resolution of content-word ambiguity by the TIE routines; i.e. possible referents are substituted in turn in the pronoun place, the parsing is done and the highest density parsing points to a preferred referent.

As we have seen in the introduction, these two processes will not resolve all anaphoric reference problems. The extended inference mode (Wilks [2]) will then handle remaining ambiguities.

CHAPTER IV

THE GENERATION ROUTINES

Translating into French requires the addition of generation patterns called STEREOTYPES. Those patterns are attached to English words in the dictionary, both to keys and content words, and carried into the IR by the analysis.

A content word has a list of stereotypes attached to each of its formulas. When a word-sense is selected during analysis, this list is carried along with the formula inside the IR. Thus, for translation purposes, the IR is not made out simply of formulas but of SENSE-PAIRS. A sense-pair is :

<formula> <list of stereotypes>

As for keys, we have seen in the last section that each key paraplate contains a stereotype, which gets attached to the template if the corresponding paraplate has been selected by the TIE routines. This stereotype is the generation rule to be used for the current fragment and possibly some of its sequents.

STEREOTYPES

The simplest form of a stereotype is a French word or phrase standing for the translation of the English word in the context. With the nouns is a gender marker. For example:

private (a soldier)	: (MASC simple soldat)
odd (for a number)	: (impair)
build	: (construire)
brandy	: (FEMI eau de vie)

Note that after processing by the analysis routines, all words are already disambiguated. Several stereotypes attached to a formula do not correspond to different senses of the source word, but to the different French constructions it can yield.

Complex stereotypes are strings of French words and functions. The functions are functions of the interlingual context of the sense-pair and evaluate to a string of French words, a blank, or to NIL. I.e. such stereotypes are CONTEXT-SENSITIVE RULES which check upon, and generate from, the sense-pair and its context, and this means other fragments as well as the current one.

When a function in a content word stereotype evaluates to NIL, then the whole stereotype fails and the next one in the list is tried.

For example, here are the two stereotypes adjoined to the ordinary sense of "advise":

(conseiller (PREOB a MAN))

(conseiller)

The first stereotype would be for translating "I advised my children to leave". The analysis routines would have matched the bare template MAN-TELL-MAN on the word triple I-advised-children. The function PREOB looks at whether the object formula of the template, i.e. the one for "children" in our example, refers to a human being; if so it generates a prepositional group with the French preposition "a", using the object sense-pair and its qualifier list. Here this yields "a mes enfants" , and the value of the whole stereotype is "conseiller a mes enfants".

For the sentence "I advise patience", whose translation might be "je conseille la patience", this stereotype would fail, as the object head is STATE. The second is simply "(conseiller)", because no prescription on how to translate the object needs to be attached to "conseiller" when the semantic object goes into a French direct object, as this is done automatically by the higher level function which constructs French clauses.

Thus we see that content words have complex stereotypes prescribing the translation of their context, when they govern an "irregular" construction, that is irregular by comparison to a set of rules matching the French syntax on the IR.

The stereotype for a content word can prescribe the translation of fragments other than the one in which it is included. A generation rule for a fragment usually comes from some key paraplate. A list of key paraplates reflects the fact that rules of syntax are usually based on some semantic classification; i.e. for given semantic categories and relationships in the context of the key, the output syntax is represented by the adjoined stereotype. However, in any natural language there will be exceptions to any classification scheme. Exceptions are dealt with here by attaching the replacement generation rule to the word governing the construction (usually the mark of the fragment).

For example, the paraplates for "to" as in "John told him / to leave", state that if the mark is an act of verbal communication (formula head TELL), then the "to" phrase should be translated by "de" followed by an infinitive: "John lui a dit de partir". This is generally the case; however "to urge", when going into "exhorter", has been coded with a TELL head, but gives the construction "a partir". Thus one of its stereotypes indicates that the construction following "exhorter" must be "a partir", while the function supervising the execution of stereotypes ensures that "a partir" will supersede "de partir", the construction which the key stereotype

attached to the template by TIE would have generated. This stereotype is as follows:

(exhorter (DIROB MAN) (FIND-LINK GOAL IR-VP) a (INFVP))

which would apply in the example:

fragment / / bare template	key	mark	case	stereotype
The delegate urged the women MAN TELL MAN	NIL	NIL	NIL	((INDCL))
who were striking MAN NOTDO DTHIS	who	(workers)	SPEC	((WHCL))
to be patient DTHIS BE KIND	to	(urged)	GOAL	(de (INFVP))

In the stereotype above, DIROB constructs a direct object with the template object if it is a human being.

FIND-LINK takes as arguments a case, and a descriptor of template types, here IR-VP, which indicates the set of templates with a dummy subject. It searches the Interlingua down from where "urged" occurs, for a fragment with case and template type according to the arguments, and with this occurence of "urged" itself as a mark. The third fragment in our example fulfills these conditions. The control function supervising the evaluation of stereotype starts then generating from it, using the piece of stereotypes which follows FIND-LINK, i.e. "a (INFVP)", instead of the stereotype of "to" which had been selected during TIE (namely "de (INFVP)").

INFVP generates an infinitive verb-phrase, after inferring its implicit subject (here women) from the semantics. Acts of verbal communication involving an attempt to influence the interlocutor, such as : persuade, order, advise , ... contain a rightmost subformula (FORCE TELL) and the subject of the dependant "to" phrase is their object. The knowledge of the implicit subject is necessary to proper agreement in French. Thus the translation of the phrase here is: "a etre patientes" where "patientes" agrees with "les femmes".

THE GENERATION PROCEDURE

The general form of the generation program is a recursive evaluation of the functions contained in stereotypes. Thus, depending on its context of occurrence, a particular word of the French output sentence may have its origin in stereotypes of different levels: content word stereotype, key word stereotype or stereotypes that are

part of a se of top level basic functions.

Key stereotypes contain top level functions which will generate French clauses and prepositional phrases, using the template to which the stereotype is attached and possibly some of its sequents. The most frequently encountered functions are:

(PREOB <French preposition>)

This will generate a prepositional group, using for the object the stereotypes attached to the object formula of the template. It calls the basic function NOUN-GROUP, which uses a sense-pair and a list of qualifying sense-pairs to generate a French nominal group.

(INDCL)

Generates a French clause in the indicative mood, from a agent-action-object triple in the IR. Given the process of fragmenting by key-word, these three elements are sometimes in different fragments and then the mark and case make explicit their relationships (the cases used are PRED (predicate) and OBJE (object)). INDCL calls the basic function CLAUSE-GROUP.

To describe the operation of CLAUSE-GROUP and NOUN-GROUP, it is necessary to introduce the two functions which handle stereotypes.

$MAP takes a stereotype as argument. It goes down the its string, building a French string in the process, by concatenating the French words and the result of evaluating the functions. It stops and returns NIL whenever one of these functions returns NIL; otherwise it returns the French string constructed. $MAP has also a feature, described below, which permits the reordering of stereotype strings.

$SELECT takes as argument a list of stereotypes and applies $MAP to each of its members in turn, until $MAP returns a non-NIL value.

The bodies of the two main syntactical functions CLAUSE-GROUP and NOUN-GROUP consist of the application of $SELECT to a list of stereotypes which reads somewhat like the phrase structure rules of the corresponding French syntactical constituent. The bottom level functions call recursively $SELECT to work on the list of stereotypes of a given content word and operate transformations on its output for proper concord, agreement, etc... To that effect, special variables carry along information about gender, number, person etc...

In fact each function in a stereotype calls $SELECT to work on a list of other stereotypes so that the sequence of $SELECT calls during execution follows the underlying tree structure of the constituent. French words found in stereotypes correspond to the terminal nodes. Generation proceeds from left to right. Concatenation to the right is done by MAP$.

However some complexity arises from the fragmented structure of the IR, and with the problem of integrating complex - context-sensitive stereotypes.

Translating fragment by fragment and preserving the interlingual order of fragments is inadequate as exemplified by:

John said a word / to him.
→ Jean lui dit un mot.

and:

the man / with blue eyes / was told / to leave.
→ on dit a l'homme aux yeux bleus de partir.

Thus, the generation rules of CLAUSE-GROUP and NOUN-GROUP must take care to pick stereotypes in the IR in an order ensuring a correct output translation, moving from template to template in the process if necessary. While evaluating stereotypes, the program maintains a cursor which points to the fragment which is being generated from. The purpose of certain functions in stereotypes (such as FIND-LINK above) is to move the cursor up and down in the IR.

Inserting complex stereotypes in the procedure poses two problems: first, when evaluated in certain contexts, a stereotype string has to be reordered. Consider:

I often urged him to leave. → Je l'ai souvent exhorte a partir.

The stereotype of "urge" applicable here is:

(exhorter (DIROB MAN) (FIND-LINK GOAL IR-VP) a (INFVP))

The value of the DIROB, namely "l'" must precede "ai exhorte" and the adverb "souvent" must be inserted between the auxiliary "ai" and "exhorte". To accomplish this, $MAP allows for the values of designated functions in a stereotype to be lifted from it and stored. Then a new string can be formed by concatenating the stored values with the values of any other function if desired, in order to produce the desired output.

Second, we need the implemention of a system of priorities for regulating the choice of generation rules. Since any word or key can dictate the output syntax for a given piece of IR, there may arise conflicts, which are resolved by having carefully settled priorities. The general idea is that a more specific rule has priority over a more general one.

Thus, when a content word stereotype (normally more specific) prescribes the translation of fragments other than its immediate context, it has priority over any key stereotype (normally more

general). As we have seen, in the example "The delegate urged the women...", generation will proceed from the stereotype of "urge" and ignore the stereotype (de (INFVP)) attached to the third fragment by the TIE routines.

CLAUSE-GROUP has a general rule for the object of an action namely concatenate the value of NOUN-GROUP applied to it. However this is overruled whenever the action stereotype dictates a different handling of the object.

A function REPHRASE allows us complex rephrasings, such as the following example: "John nearly killed himself", which translates properly into "John a failli se tuer", i. e. the adverb "nearly" goes into the verb "faillir". "Nearly" has the following stereotype:

((REPHRASE VERB-GROUP ((VERB-GROUP FAILLIR)(INFVG0)))

The function REPHRASE indicates that the execution of the function VERB-GROUP - a constituent in CLAUSE-GROUP - should be replaced by the evaluation of the stereotype which is its second argument. This will generate a verb-group constructed from "faillir", followed by an infinitive verb-group with the "current" subject (that of "faillir") as its own subject. Any stereotype from a REPHRASE call takes precedence over whatever stereotypes the substituted function contained.

Implementation of these priorities requires some functions in the stereotypes to test other stereotypes in advance in order to decide what to generate next. And the overall control function does some book-keeping; i. e. it keeps track of which sense-pair and fragments have already been generated from, and which stereotype it used.

The overall control function sets the cursor to the first fragment and picks up its stereotype; $MAP is run though it, and the cursor moves up or down in the IR as the recursive structure calls for. When $MAP pops up, after exhaustion of the first stereotype, the French phrase that is its value is concatenated to the text already generated. The program then moves down into the IR until it finds a fragment which has not been translated yet; the process is then reiterated as with the first fragment.

The generation procedure is formally equivalent to an augmented recursive transition network (Woods (?)). Functions in stereotypes correspond to the syntactical constituents on the arcs. A list of stereotypes as an argument for $EVAL corresponds to several arcs leaving from a given state. Stereotypes may include predicates which play the role of Woods' tests: the result of their evaluation determine whether an arc will be followed or not. Woods' registers take the form of LISP PROG variables, which function as pushdown stacks and and hold pieces of generated text or any desired information.

References

Wilks, Y.: "Preference Semantics", Stanford A.I. Project Memo #206, 1973. To appear in (ed.Keenan) The Formal Semantics of Natural Language. Cambridge U.P.

Wilks, Y.: "Natural language inference", Stanford A.I. Project Memo #211, 1973.

Woods, W.A.: "Augmented transition networks for natural language analysis", Report # CS-1, Aiken Computation Laboratory, Harvard University, Dec.,1969.